

AMENDMENTS TO THE CLAIMS

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

- 1 1. (Currently amended) A method for generating code to perform
2 anticipatory prefetching for data references, comprising:
3 receiving code to be executed on a computer system;
4 analyzing the code to identify data references to be prefetched; and
5 inserting prefetch instructions into the code in advance of the identified
6 data references, wherein inserting prefetch instructions involves inserting multiple
7 prefetch instructions for a given cache line, and wherein inserting the prefetch
8 instructions involves,
9 attempting to calculate a stride value for a given data
10 reference within a loop,
11 if the stride value cannot be calculated, setting the stride
12 value to a default stride value, and
13 inserting a prefetch instruction to prefetch the given data
14 reference for a subsequent loop iteration based on the stride value;
15 wherein the stride value is constant for some $[()]$ but not necessarily
16 all $[()]$ loop iterations.
- 1 2. (Original) The method of claim 1, further comprising allowing a system
2 user to specify the default stride value.

1 3. (Original) The method of claim 1, wherein calculating the stride value
2 involves:
3 identifying an induction variable for the stride value;
4 identifying a stride function for the stride value; and
5 calculating the stride value based upon the stride function and the
6 induction variable.

1 4. (Original) The method of claim 1, wherein inserting the prefetch
2 instruction based on the stride value involves:
3 calculating a prefetch cover distance by dividing a cache line size by the
4 stride value;
5 calculating a prefetch ahead distance as a function of a prefetch latency,
6 the prefetch cover distance and an execution time of a loop; and
7 calculating a prefetch address by multiplying the stride value by the
8 prefetch cover distance and the prefetch ahead distance and adding the result to an
9 address accessed by the given data reference.

1 5. (Original) The method of claim 1, wherein analyzing the code involves:
2 identifying loop bodies within the code; and
3 identifying data references to be prefetched from within the loop bodies.

1 6. (Original) The method of claim 5, wherein analyzing the code to
2 identify data references to be prefetched involves examining a pattern of data
3 references over multiple loop iterations.

1 7. (Original) The method of claim 1, wherein analyzing the code involves
2 analyzing the code within a compiler.

1 8. (Currently amended) A computer-readable storage medium storing
2 instructions that when executed by a computer cause the computer to perform a
3 method for generating code to perform anticipatory prefetching for data
4 references, the method comprising:
5 receiving code to be executed on a computer system;
6 analyzing the code to identify data references to be prefetched; and
7 inserting prefetch instructions into the code in advance of the identified
8 data references, wherein inserting prefetch instructions involves inserting multiple
9 prefetch instructions for a given cache line, and wherein inserting the prefetch
10 instructions involves,
11 attempting to calculate a stride value for a given data
12 reference within a loop,
13 if the stride value cannot be calculated, setting the stride
14 value to a default stride value, and
15 inserting a prefetch instruction to prefetch the given data
16 reference for a subsequent loop iteration based on the stride value;
17 wherein the stride value is constant for some $[[()]]$ but not necessarily all $[[()]]$
18 loop iterations.

1 9. (Original) The computer-readable storage medium of claim 8, wherein
2 the method further comprises allowing a system user to specify the default stride
3 value.

1 10. (Original) The computer-readable storage medium of claim 8, wherein
2 calculating the stride value involves:
3 identifying an induction variable for the stride value;
4 identifying a stride function for the stride value; and

5 calculating the stride value based upon the stride function and the
6 induction variable.

1 11. (Original) The computer-readable storage medium of claim 8, wherein
2 inserting the prefetch instruction based on the stride value involves:

3 calculating a prefetch cover distance by dividing a cache line size by the
4 stride value;

5 calculating a prefetch ahead distance as a function of a prefetch latency,
6 the prefetch cover distance and an execution time of a loop; and

7 calculating a prefetch address by multiplying the stride value by the
8 prefetch cover distance and the prefetch ahead distance and adding the result to an
9 address accessed by the given data reference.

1 12. (Original) The computer-readable storage medium of claim 8, wherein
2 analyzing the code involves analyzing the code within a compiler.

1 13. (Currently amended) An apparatus that generates code to perform
2 anticipatory prefetching for data references, comprising:

3 a receiving mechanism that is configured to receive code to be executed on
4 a computer system;

5 an analysis mechanism that is configured to analyze the code to identify
6 data references to be prefetched; and

7 an insertion mechanism that is configured to insert prefetch instructions
8 into the code in advance of the identified data references wherein the insertion
9 mechanism facilitates inserting multiple prefetch instructions for a given cache
10 line;

11 wherein the insertion mechanism is configured to,

12 attempt to calculate a stride value for a given data reference
13 within a loop,
14 set the stride value to a default stride value if the stride
15 value cannot be calculated, and to
16 insert a prefetch instruction to prefetch the given data
17 reference for a subsequent loop iteration based on the stride value;
18 wherein the stride value is constant for some $[[()]]$ but not necessarily all $[[()]]$
19 loop iterations.

1 14. (Original) The apparatus of claim 13, further comprising a
2 configuration mechanism that is configured to receive the default stride value
3 from a system user.

1 15. (Original) The apparatus of claim 13, wherein while calculating the
2 stride value, the insertion mechanism is configured to:
3 identify an induction variable for the stride value;
4 identify a stride function for the stride value; and to
5 calculate the stride value based upon the stride function and the induction
6 variable.

1 16. (Original) The apparatus of claim 13, wherein the insertion mechanism
2 is configured to:
3 calculate a prefetch cover distance by dividing a cache line size by the
4 stride value;
5 calculate a prefetch ahead distance as a function of a prefetch latency, the
6 prefetch cover distance and an execution time of a loop; and to

7 calculate a prefetch address by multiplying the stride value by the prefetch
8 cover distance and the prefetch ahead distance and adding the result to an address
9 accessed by the given data reference.

1 17. (Original) The apparatus of claim 13, wherein the apparatus resides
2 within a compiler.

1 18-45 (Canceled).